

Aplikasi Pengenalan Ucapan Dengan Jaringan Syaraf Tiruan Propagasi Balik Untuk Pengendalian Robot Bergerak

Mahmud Irfandy (L2F 004 490)

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro,
Semarang, Indonesia
rfun_2004@yahoo.co.id

Abstrak – *Pengenalan ucapan dengan jaringan syaraf tiruan dapat dipahami dan dimengerti melalui beberapa metode ekstraksi ciri, diantaranya dengan metode LPC dan transformasi Fourier. Linear Predictive Coding adalah salah satu alat dalam pemrosesan atau analisa sinyal untuk memperoleh fitur atau ciri unik dari tiap pola suara. Sedangkan Transformasi Fourier digunakan untuk memperjelas ciri tiap pola karena dapat memberikan informasi-informasi yang disajikan dalam ranah frekuensi baik yang bersifat diskret maupun kontinu.*

JST (Jaringan Syaraf Tiruan) banyak digunakan untuk aplikasi pengenalan pola (pattern recognition). Kemampuan untuk pembelajaran dari data pelatihan dan generalisasi ke situasi/kondisi yang baru adalah alasan mendasar mengapa JST banyak digunakan.

Hasil yang di harapkan adalah suatu sistem kendali robot yang efektif hanya dengan menggunakan suara manusia. Suara manusia yang masuk akan dikenali oleh JST dan diubah menjadi suatu karakter ASCII yang dapat dikenali oleh robot kemudian dikirimkan melalui komunikasi serial ke robot.

*Dalam aplikasi ini pengenalan ucapan meliputi lima kata, diantaranya maju, mundur, kanan, kiri dan stop. Mula-mula sinyal suara direkam dalam bentuk file *.wav. Kemudian sinyal suara analog dicuplik menjadi sinyal digital dengan kecepatan cuplik 8000 Hz. Untuk proses ekstraksi parameter suara digunakan metode Linear Predictive Coding (LPC) untuk mendapatkan koefisien cepstral. Koefisien cepstral LPC ini ditransformasikan ke dalam domain frekuensi dengan beberapa titik pada Fast Fourier Transform (FFT), yaitu 32, 64, 128, dan 512 point. Selanjutnya dari data tersebut diambil setengahnya saja yang sudah mewakili representasi frekuensi dari sinyal tersebut. Data numeris tersebut dilatih dengan Jaringan Syaraf Tiruan untuk dikenali. Metoda pembelajaran yang digunakan adalah Back Propagation. Dari hasil pembelajaran akan didapatkan bobot jaringan yang telah berubah. Jaringan diuji dengan memberikan masukan berupa data pengujian.*

Hasil pengujian menunjukkan bahwa pengenalan jaringan terhadap data baru lebih rendah terhadap data latihan. Dari beberapa variasi pengujian diperoleh jaringan terbaik net9 dengan prosentase pengenalan suara dari responden di dalam database sebesar 86 %, dan prosentase pengenalan suara dari responden di luar database sebesar 66,67 %.

Kata Kunci : *Pengenalan ucapan, LPC, FFT, JST, Back Propagation, serial*

I. Pendahuluan

Pengolahan sinyal merupakan hal yang sangat penting dalam ilmu pengetahuan dan teknologi khususnya dalam teknologi komunikasi baik dalam pengolahan sinyal analog maupun pengolahan sinyal digital. Salah satu bidang pengolahan sinyal yang sangat berpengaruh dalam teknologi komunikasi adalah pengenalan ucapan (*speech recognition*).

1.1 Latar Belakang

Teknologi pengenalan ucapan saat ini telah mengalami perkembangan yang cukup pesat. Banyak aplikasi-aplikasi yang telah dihasilkan seiring dengan metode dan proses dari pengenalan ucapan tersebut. Dari perkembangan teknologi pengolahan sinyal suara ini munculah ide untuk membuat suatu sistem kendali robot yang efektif hanya dengan menggunakan suara manusia.

Dalam *speech recognition* terdapat beberapa permasalahan yang sering muncul dan berbeda untuk setiap permasalahannya misalnya kecepatan pengenalannya, ketepatan pengenalannya, dan lain sebagainya. Untuk itu diperlukan suatu algoritma

pembelajaran mesin yang dapat digunakan untuk melakukan ekstraksi pola atau ciri suara dengan mempelajari ciri-ciri suara sebelumnya.

Algoritma jaringan syaraf tiruan merupakan salah satu algoritma pembelajaran mesin yang meniru proses yang terjadi pada kehidupan alami. Algoritma jaringan syaraf tiruan merupakan algoritma yang memungkinkan mesin untuk belajar dari pengalaman. Algoritma ini cocok untuk diaplikasikan ke permasalahan yang hanya diketahui masukan serta keluarannya, dan proses antara masukan dan keluaran tidak diketahui. Dengan alasan tersebut, algoritma jaringan syaraf tiruan ini dapat dipergunakan untuk menyelesaikan permasalahan pengenalan suara pada sistem ini.

1.2 Tujuan

Tujuan dari tugas akhir ini adalah membuat aplikasi pengenalan suara untuk pengendalian robot dengan menggunakan jaringan syaraf tiruan sebagai algoritma pembelajaran dan pemodelan dalam pengenalan suara.

1.3 Batasan Masalah

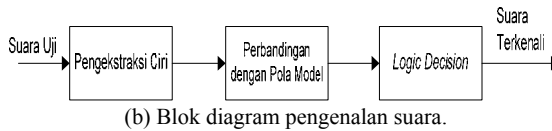
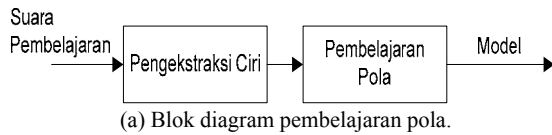
Agar pembahasan atau analisis tidak melebar dan terarah, maka permasalahan dibatasi pada :

1. Data masukan berupa 5 buah kata bahasa Indonesia, yaitu maju, mundur, kanan, kiri, dan stop yang diambil dari 6 orang responden yang terdiri dari 5 orang pria dan 1 orang wanita. Masing-masing responden diambil datanya sebanyak 10 buah contoh bunyi untuk setiap kata.
2. Analisis sinyal dilakukan dengan ekstraksi ciri menggunakan LPC (*Linear Predictive Coding*) dan FFT (*Fast Fourier Transform*).
3. Proses pengenalan suara dilakukan dengan Jaringan Syaraf Tiruan *Multilayer Perceptron* dengan metode pembelajaran *Back Propagation*.
4. Proses pengiriman data dari komputer melalui komunikasi serial.
5. Pembuatan program dilakukan dengan menggunakan bantuan bahasa pemrograman Matlab 6.5.

II. Landasan Teori

2.1 Pengenalan Suara

Salah satu bentuk pendekatan pengenalan suara adalah pendekatan pengenalan pola yang terdiri dari dua langkah yaitu pembelajaran pola suara dan pengenalan suara melalui perbandingan pola. Tahap perbandingan pola adalah tahap saat suara yang akan dikenali dibandingkan polanya dengan setiap kemungkinan pola yang telah dipelajari dalam fase pembelajaran, untuk kemudian diklasifikasikan dengan pola terbaik yang cocok. Blok diagram pembelajaran pola dan pengenalan suara ditunjukkan pada gambar di bawah ini.



Gambar 1. Blok diagram pembelajaran pola dan pengenalan suara.

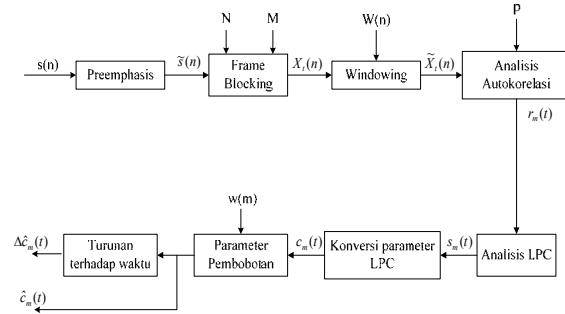
Pengenalan suara secara umum dapat dibagi menjadi tiga tahap, yaitu tahap ekstraksi ciri, tahap pemodelan atau pembelajaran, dan tahap pengenalan suara.

2.1.1 Ekstraksi Ciri dengan Metode LPC

Salah satu metode yang digunakan untuk proses ekstraksi ciri adalah *Linear Predictive Coding* (LPC). Prinsip dasar dari pemodelan sinyal dengan menggunakan LPC adalah bahwa contoh sinyal ucapan $s(n)$ pada waktu ke- n dapat diperkirakan sebagai kombinasi linier p sampel sinyal ucapan sebelumnya yaitu

$$s(n) \approx a_1 s(n-1) + a_2 s(n-2) + \dots + a_p s(n-p)$$

dengan koefisien a_1, a_2, \dots, a_p diasumsikan bernilai konstan pada *frame* analisis ucapan. Prosedur untuk mendapatkan koefisien LPC diperlihatkan pada Gambar 2.



Gambar 2. Blok diagram analisis LPC.

2.1.3 Algoritma Transformasi Fourier Cepat (*Fast Fourier Transform*)

Untuk lebih memudahkan dalam perhitungan, transformasi Fourier diskret selanjutnya dapat dikembangkan dengan cara mengefisienkan perhitungan juga sifat simetri dari faktor fasa W_N yang menjadi sifat transformasi Fourier diskret dengan tujuan mengurangi jumlah perhitungan yang dibutuhkan.

Untuk lebih memudahkan dalam manipulasi atau perhitungan penjumlahan,

$$x(t) = \frac{ao}{2} + \sum_{k=1}^{\infty} (a_k \cos(\omega kt) + b_k \sin(\omega kt))$$

dapat dituliskan kembali dalam bentuk bilangan kompleks eksponensial berdasarkan identitas Euler :

$$\cos(\theta) = \frac{e^{j\theta} + e^{-j\theta}}{2}, \sin(\theta) = \frac{e^{j\theta} - e^{-j\theta}}{2j}$$

Sehingga,

$$x(t) = ao + \sum_{k=1}^{\infty} (a_k \frac{e^{j\omega kt} + e^{-j\omega kt}}{2} + b_k \frac{e^{j\omega kt} - e^{-j\omega kt}}{2j})$$

$$= ao + \sum_{k=1}^{\infty} (\frac{1}{2} (a_k - b_k) e^{j\omega kt} + \frac{1}{2} (a_k + b_k) e^{-j\omega kt})$$

dinyatakan $y_n = 1/2 (a_n - j b_n)$, dapat kita gunakan fakta-fakta bahwa a_n dan b_n adalah koefisien Fourier fungsi periodik, dengan fungsi kosinus adalah fungsi periodik genap, dan fungsi sinus adalah fungsi periodik ganjil.

Algoritma FFT (*Fast Fourier Transform*) yang banyak digunakan secara luas adalah algoritma dengan bilangan dasar 2, artinya masukan yang dibutuhkan sebanyak $N = 2^V$. Hal yang pertama dilakukan dalam pengembangan algoritma FFT (*Fast Fourier Transform*) dengan bilangan dasar 2 ialah dengan menyederhanakan penulisan persamaan DFT (*Discrete Fourier Transform*) dan juga membuat beberapa hubungan matematis, sehingga persamaan (2.19) dapat ditulis kembali menjadi :

$$X_1(k) = \sum_{n=0}^{N-1} x_n e^{-j2\pi nk/N}, k = 0, 1, \dots, N-1$$

Dan juga faktor $e^{-j2\pi n/N}$ akan ditulis sebagai W_N , maka :

$$W_N = e^{-j2\pi/N}$$

Sehingga persamaan (2.27) menjadi :

$$X_1(k) = \sum_{n=0}^{N-1} x_n W_N^{kn}, k = 0, 1, \dots, N-1$$

Akan sangat bermanfaat pada saat ini untuk memperhatikan hubungan yang melibatkan W_N pertama

$$W_N^2 = (e^{-j2\pi/N})^2 = e^{-j2\pi 2/N} = e^{-j2\pi(N/2)} = W_{N/2}$$

Kedua

$$\begin{aligned} W_N^{(K+N/2)} &= W_N^K W_N^{N/2} = W_N^K e^{-j(2\pi/N)(N/2)} \\ &= W_N^K e^{-j\pi} (\cos \pi - j \sin \pi) \\ &= -W_N^K \end{aligned}$$

bila hasilnya disimpulkan yang berhubungan dengan W_N maka kita dapatkan

$$W_N = e^{-j2\pi/N}$$

$$W_N^2 = W_{N/2}$$

$$W_N^{(K+N/2)} = -W_N^K$$

Prosedur ini dijelaskan sebagai berikut : sufiks n , dalam persamaan (2.28) terletak dari $n = 0$ hingga $n = N-1$, berhubungan dengan nilai data $X_0, X_1, X_2, X_3, \dots, X_{N-1}$. Runtun nomor genap adalah $X_0, X_2, X_4, X_6, \dots, X_{N-2}$ dan runtun bernomor ganjil adalah $X_1, X_3, X_5, \dots, X_{N-1}$, kedua runtun mengandung $N/2$ poin. Istilah dalam runtun genap dapat diwakili dengan X_{2n} dengan $n = 0$ hingga $n = N/2 - 1$ sementara untuk runtun ganjilnya menjadi X_{2n+1} , maka persamaan (2.28) dapat ditulis ulang menjadi :

$$\begin{aligned} x_1(k) &= \sum_{n=0}^{N/2-1} x_{2n} W_N^{2nK} + \sum_{n=0}^{N/2-1} x_{2n+1} W_N^{(2n+1)k} \\ &= \sum_{n=0}^{N/2-1} x_{2n} W_N^{2nK} + W_N^K \sum_{n=0}^{N/2-1} x_{2n+1} W_N^{2nK}, k = 0, 1, \dots, N-1 \end{aligned}$$

dengan menggunakan persamaan (2.31b) memberikan

$$W_N^{2nK} = W_{N/2}^{nk} \text{ sehingga persamaan (2.32) menjadi :}$$

$$x(k) = \sum_{n=0}^{N/2-1} x_{2n} W_{N/2}^{nk} + W_N^K \sum_{n=0}^{N/2-1} x_{2n+1} W_{N/2}^{nk}, k = 0, 1, \dots, N-1$$

persamaan (2.33) dapat ditulis menjadi :

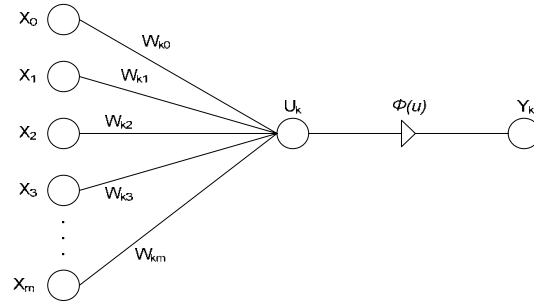
$$X_1(k) = X_{11}(k) + W_N^K X_{12}(k), k = 0, 1, \dots, N-1$$

Bila dibandingkan persamaan (2.28) dan persamaan (2.34) terlihat bahwa $X_{11}(k)$ sesungguhnya adalah transformasi Fourier diskret runtun genap, sementara $X_{12}(k)$ merupakan runtun ganjil. Karenanya seperti yang sudah disebutkan bahwa transformasi Fourier diskret : $X_{11}(k)$ dan $X_{12}(k)$ faktor $W_{N/2}^K$ terdapat di kedua $X_{11}(k)$ dan $X_{12}(k)$ dan hanya perlu dihitung sekali saja.

2.1.3 Pemodelan Jaringan Syaraf Tiruan

Model matematis dari neuron dapat dilihat pada Gambar 3. Masukan dan bobot merupakan model matematis dari dendrit, penjumlahan serta fungsi pengaktif merupakan model matematis dari soma

(badan sel), serta keluaran merupakan model matematis dari akson.



Gambar 3. Model matematis neuron

Setiap jalur nilai masukan ke neuron mempunyai bobot, dan terdapat satu jalur masukan tambahan yang bukan merupakan jalur masukan, yaitu jalur ke-0. Jalur ke-0 ini dapat berfungsi sebagai nilai koreksi atau dapat juga berfungsi sebagai nilai *threshold* (nilai ambang batas). Impuls pada sistem syaraf manusia dimodelkan sebagai nilai yang dimasukkan melalui jalur nilai masukan ke dalam model neuron. Nilai pada setiap jalur masukan akan dikalikan dengan bobot jalur, dan kemudian dijumlahkan. Nilai yang dihasilkan dari penjumlahan seluruh masukan tersebut dapat berada pada *range* yang sangat besar sehingga akan mempersulit pengolahan selanjutnya. Untuk menghindari hal tersebut, nilai hasil penjumlahan akan diproses terlebih dulu dengan menggunakan sebuah fungsi φ (phi) yang disebut fungsi aktivasi. Penghitungan nilai hasil dapat dituliskan secara matematis sebagai berikut:

$$y_k = \varphi(u)$$

$$y_k = \varphi\left(\sum_{i=0}^m x_i w_{ki}\right)$$

Proses pembelajaran pada suatu neuron dilakukan dengan memperbarui nilai bobot setiap masukan (w_i) sesuai dengan pembelajaran yang diberikan. Bentuk data pembelajaran adalah berupa pasangan vektor nilai masukan ke neuron dan nilai target keluaran (t). Nilai masukan akan dimasukkan ke dalam neuron, dan menghasilkan nilai keluaran (y). Kesalahan dari nilai keluaran yang dihasilkan dihitung, dan nilai bobot w untuk setiap masukan diperbarui menggunakan:

$$w_i(n+1) = w_i(n) + \Delta w_i$$

$$\Delta w_i = \eta(t - y)x_i$$

Dengan η (μ) adalah kecepatan pembelajaran, yaitu seberapa besar bobot w terbaru pada setiap langkah pembelajaran. Proses ini dilakukan berulang-ulang dengan data pembelajaran yang berbeda. Pada akhirnya, bobot nilai masukan akan berisi nilai bobot yang sesuai.

2.1.4 Metode Pelatihan Backpropagation

Algoritma pelatihan *backpropagation* meliputi 3 fase. Fase pertama adalah fase maju. Pola masukan dihitung maju mulai dari *layer* masukan hingga *layer*

keluaran menggunakan fungsi aktivasi yang ditentukan. Fase kedua adalah fase mundur. Selisih antara keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan tersebut dipropagasikan mundur, dimulai dari garis yang berhubungan langsung dengan unit-unit di *layer* keluaran. Fase ketiga adalah modifikasi bobot untuk menurunkan kesalahan yang terjadi.

Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi. Umumnya kondisi penghentian yang sering dipakai adalah jumlah iterasi atau kesalahan. Iterasi akan dihentikan jika jumlah iterasi yang dilakukan sudah melebihi jumlah maksimum iterasi yang ditetapkan, atau jika kesalahan yang terjadi sudah lebih kecil dari batas toleransi kesalahan yang ditetapkan.

III. Perancangan dan Implementasi Sistem

Secara garis besar, aplikasi pengenalan sinyal ucapan ini terdiri dari dua bagian, yaitu bagian pengenalan ucapan dan pengiriman data ke robot. Sistem ini diwujudkan ke dalam suatu perangkat lunak (program) menggunakan bahasa pemrograman Matlab 6.5. Secara umum pembuatan program ini mengikuti alur sesuai yang ditunjukkan pada Gambar 5 berikut.



Gambar 5. Diagram alir program utama

Pembuatan sistem pengenalan ucapan terdiri dari proses pengelolaan basis data, proses ekstraksi ciri, dan proses pemodelan dan pelatihan. Ketiga proses tersebut sering disebut dengan proses pembelajaran. Proses

pengenalan ucapan merupakan proses perbandingan antara suara uji dengan model suara yang didapat dari proses pembelajaran.

Sistem yang dibuat ditujukan untuk dapat mengenali beberapa kata masukan, yaitu "maju", "mundur", "kanan", "kiri", dan "stop". Masing-masing kata mewakili satu perintah masukan pada robot. Kata "maju" untuk menggerakkan robot maju ke depan, "mundur" untuk memutar posisi robot dengan sudut 180° dan menggerakkan ke depan, "kanan" untuk menggerakkan robot belok ke kanan, "kiri" untuk menggerakkan robot belok ke kiri, dan kata "stop" untuk menghentikan robot.

3.3 Pengelolaan Basis Data

Dalam pembuatan basis data dilakukan perekaman suara dengan 6 orang responden yang terdiri dari 5 orang pria dan 1 orang wanita dengan rentang usia 22-23 tahun. Tiap responden melakukan perekaman sebanyak 10 kali untuk setiap kata, sehingga diperoleh 50 suara basis data untuk tiap responden. Database ditambah dengan perekaman suara di luar kata "maju", "mundur", "kanan", "kiri", dan "stop" yang akan dikenali sebagai "Suara Tidak Dikenali" sebanyak 30 suara, sehingga jumlah total keseluruhan basis data ialah 330 suara. Perekaman suara dalam pembuatan basis data ini dilakukan selama 1 detik untuk tiap ucapan. Pencuplikan dilakukan pada frekuensi 8000 Hz dengan resolusi 16 bit.

Kecepatan pencuplikan tersebut dilakukan dengan didasarkan asumsi bahwa sinyal percakapan (*speech*) berada pada daerah frekuensi 300-3400 Hz sehingga memenuhi kriteria Nyquist yang menyatakan bahwa frekuensi penyampling minimal 2 kali frekuensi maksimal dari sinyal yang akan disampling untuk menghindari *aliasing* :

$$f_s \geq 2f_h \quad f_h = f_{in\text{ tertinggi}}$$

3.2 Ekstraksi Ciri

Proses pengestraksian ciri merupakan proses untuk mendapatkan parameter-parameter sinyal suara. Parameter-parameter inilah yang nantinya digunakan untuk membedakan satu kata dengan kata yang lain.

Proses awal ekstraksi ciri yaitu dengan normalisasi amplitudo terlebih dahulu, setelah itu dilakukan pendeteksian awal dan akhir kata.

Normalisasi amplitudo dilakukan dengan cara membagi semua nilai input dengan nilai maksimum dari input itu sendiri. Sehingga untuk semua sinyal masukan memiliki nilai maksimum yang sama yaitu 1.

Sedangkan deteksi awal-akhir digunakan pada proses untuk mendeteksi mulai sinyal ucapan awal dan berakhir ketika sudah tidak diucapkan. Sehingga tidak disalah artikan untuk tiap sinyal yang masuk. Setelah itu sinyal terlebih dahulu disamakan jumlah datanya, sehingga pada proses analisa LPC bisa didapat jumlah frame dan panjang masing-masing frame yang sama. Walaupun tahap ini akan memperpanjang data dan proses pada jaringan syaraf tiruan, tahap ini mampu memperkecil kesalahan pengenalan kata. Terutama untuk kata yang memiliki jumlah suku kata berbeda.

Proses normalisasi amplitudo, pendeteksian awal dan akhir kata, serta penyamaan panjang data pada Matlab dilakukan dengan fungsi 'clean' sebagai berikut :

```
[majul,fs] = wavread('mahmudmajul.wav');
potong_majul = clean(majul);
```

Data dari hasil normalisasi selanjutnya melewati proses *preemphasis*. Setelah melalui proses *preemphasis*, sinyal suara dibagi dalam beberapa *frame* sepanjang *N sample* suara dengan jarak antara *frame* yang berdekatan dipisahkan oleh *M sample*. Setiap *frame* kemudian dijenkelakan menggunakan *Hamming window*, dianalisis autokorelasi, dan dianalisis LPC sehingga didapatkan koefisien LPC. Semua proses tersebut dapat dilakukan dengan satu fungsi 'lpc_cep'.

```
koef_lpc1 = lpc_cep(potong_majul);
koef_lpc1 = rerata(koef_lpc1);
```

Proses *frame blocking* yang dilakukan pada sistem ini ditetapkan tiap 30 mili detik dengan jarak antar *frame* 10 mili detik. Jadi dengan kecepatan cuplik sebesar 8000 Hz maka tiap *frame* akan berisi 240 *byte* data dengan jarak antar *frame* 80 *byte* data atau dengan kata lain *overlap* yang terbentuk sebesar 160 *byte* data. Dengan ketentuan *frame* seperti di atas, maka untuk data hasil cuplik sebanyak 5000 data maka akan terbentuk $\frac{5000-160}{80} = 60$ frame (hasil pembulatan dari 60,5).

Untuk perhitungan koefisien LPC, digunakan orde LPC 12 sehingga diperoleh data output sebanyak $(12+1) \times 60 = 780$ data. Selanjutnya koefisien LPC ini di rata-rata sehingga diperoleh hasil akhir koefisien LPC sebanyak 13×1 data.

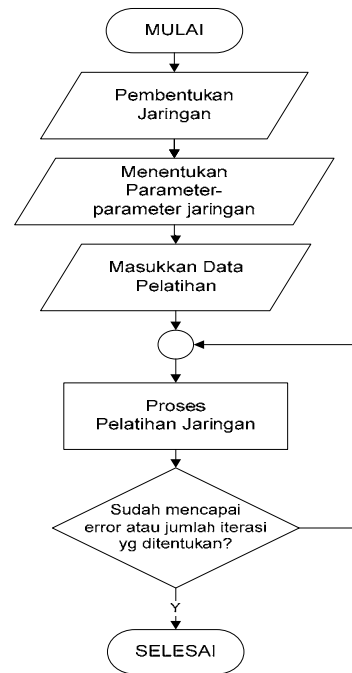
Selanjutnya koefisien LPC tersebut diproses dengan *Fast Fourier Transform* (FFT) untuk mendapatkan sinyal pada domain frekuensi. Hal ini bertujuan agar perbedaan antar pola kata yang satu dengan yang lain terlihat lebih jelas sehingga ekstraksi parameter sinyal memberikan hasil yang lebih baik.. Contoh proses FFT ditunjukkan oleh senarai program berikut :

```
fft_majul = abs(fft(koef_lpc1,512));
```

Proses *Fast Fourier Transform* (FFT) yang digunakan memakai 32, 64, 128 dan 512 *point* dan karena hasil FFT simetris maka keluaran FFT tersebut hanya diambil setengahnya saja. Dari data FFT ini kemudian dijadikan masukan bagi jaringan syaraf tiruan.

3.3 Pelatihan Jaringan

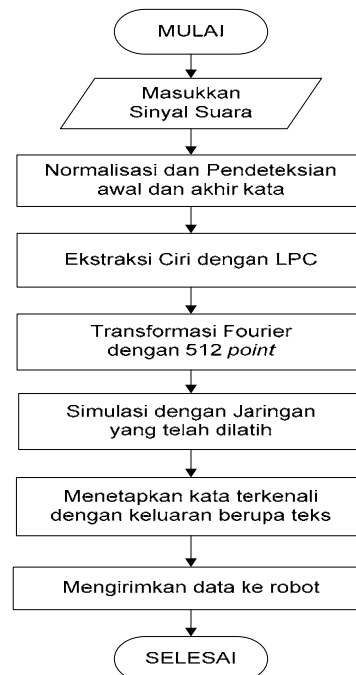
Setelah melalui tahap transformasi Fourier selanjutnya parameter yang dihasilkan dimasukkan ke dalam jaringan syaraf tiruan dengan menggunakan metode pembelajaran *Back Propagation*. Diagram alir proses pelatihan jaringan dapat dilihat pada gambar di bawah ini :



Gambar 6. Diagram alir proses pelatihan jaringan.

3.2.1 Pengenalan Ucapan

Diagram alir proses pengenalan ucapan dengan menggunakan jaringan syaraf tiruan *back propagation* ditunjukkan oleh gambar berikut :



Gambar 7 Diagram alir proses pengenalan ucapan

Sebelum mengirimkan data melalui komunikasi serial, kita perlu membuat objek serial pada program Matlab dengan menggunakan perintah sebagai berikut :

```
s1 = serial('COM1');
s1.Baudrate = 4800;
s1.DataBits = 8;
s1.StopBits = 1;
s1.Parity = 'none';
```

Dari senarai program di atas dapat kita lihat konfigurasi yang digunakan pada objek serial ini yaitu menggunakan jalur COM1, *baudrate* 4800 bps, jumlah data bit 8, dan jumlah stop bit 1. Sebelum mengirimkan data, objek serial ini harus dibuka terlebih dahulu dengan perintah `fopen(s1)`. Setelah itu data kita kirimkan dengan perintah `fwrite(s1, '8', 'char')` misalnya untuk memerintahkan robot bergerak maju. Untuk menutup port serial digunakan perintah `fclose(s1)`.

IV. Pengujian dan Analisis

4.1 Pengujian

Hal-hal yang dibahas dalam pengujian dan analisis tugas akhir ini adalah pengaruh jumlah *hidden neuron*, besarnya laju pembelajaran, jumlah *hidden layer* jaringan, serta jumlah titik pada proses FFT.

Analisis terbagi menjadi 2, yaitu analisis tahap pelatihan, dan analisis tahap pengenalan.

4.1.1 Analisis Tahap Pelatihan

Dari penelitian diperoleh hasil pelatihan dengan beberapa variasi parameter pengujian sebagai berikut:

Tabel 1. Hasil perbandingan jumlah hidden neuron

Nama	Struktur net	(<i>mu</i>)	iterasi	MSE akhir
net1	(150, 50)	0,01	2000	0,00431053
net2	(250, 100)	0,01	2000	0,000985151
net3	(350, 150)	0,01	1025	$9,94634 \times 10^{-21}$

Tabel 2. Hasil perbandingan laju pembelajaran

Nama	Struktur net	(<i>mu</i>)	iterasi	MSE akhir
net4	(250, 100)	0,01	2000	0,000985151
net5	(250, 100)	0,1	853	$9,90568 \times 10^{-21}$
net6	(250, 100)	0,25	2000	0,00501992

Tabel 3. Hasil perbandingan jumlah hidden layer

Nama	Struktur net	(<i>mu</i>)	iterasi	MSE akhir
net7	(350, 150)	0,1	885	$9,7462 \times 10^{-21}$
net8	(350, 200, 50)	0,1	724	$9,56501 \times 10^{-21}$
net9	(350, 250, 150, 50)	0,1	676	$8,84529 \times 10^{-21}$

Dari beberapa hasil di atas didapatkan bahwa semakin besar jumlah hidden neuron maka jumlah iterasi pelatihan jaringan semakin kecil dan jaringan dapat mencapai MSE yang diharapkan. Laju pembelajaran terbaik yang didapatkan yaitu sebesar 0,1. Dari hasil tersebut didapatkan pula bahwa semakin besar jumlah hidden layer maka iterasi pelatihan yang dibutuhkan juga semakin kecil.

4.1.2 Analisis Tahap Pengenalan

Pada pengujian ini digunakan 5 orang responden, 3 orang dari luar database, dan 2 orang dari dalam database. Selanjutnya dari kelima responden itu direkam masing-masing 10 pengucapan untuk tiap kata, sehingga total ada 250 data yang diujikan. Dari hasil pengujian diperoleh hasil sebagai berikut :

Tabel 4. Hasil pengenalan tiap jaringan dengan responden di dalam database

Jaringan	Prosentase Pengenalan %
net1	76 %
net2	75 %
net3	81 %
net4	75 %
net5	80 %
net6	71 %
net7	80 %
net8	82 %
net9	86 %

Tabel 5. Hasil pengenalan tiap jaringan dengan responden di luar database

Jaringan	Prosentase Pengenalan %
net1	45,33 %
net2	44 %
net3	58 %
net4	44 %
net5	58 %
net6	49,33 %
net7	65,33 %
net8	65,33 %
net9	66,67 %

Dari 2 tabel di atas dapat kita lihat bahwa prosentase pengenalan dengan responden di luar database lebih kecil daripada prosentase pengenalan dengan responden di dalam database. Hal ini disebabkan oleh adanya beberapa faktor, diantaranya faktor derau, karakteristik sinyal suara yang dikenali, dan adanya pengaruh distorsi

4.1.3 Pengujian Jumlah Titik Pada FFT

Jumlah titik yang digunakan dalam pengujian ini adalah 32, 64, 128 dan 512 titik. Hasil percobaan dapat kita lihat pada tabel berikut.

Tabel 6. Hasil pengujian variasi jumlah titik FFT

Jaringan	Jumlah titik FFT	Prosentase Pengenalan %
Net_fft_32	32	68 %
Net_fft_64	64	72 %
Net_fft_128	128	71,2 %
Net_fft_512	512	74,4 %

Dari hasil di atas diperoleh bahwa hasil FFT dengan 512 *point* memiliki prosentase pengenalan kata tertinggi yaitu sebesar 74,4 %.

4.1.4 Pengujian Dengan Respon Robot

Setelah melakukan pengujian pada sistem pengenalnya saja, selanjutnya adalah melakukan pengujian untuk di komunikasikan dengan robot. Untuk pengujian ini yang diutamakan adalah pengukuran waktu *delay* yang di hitung mulai dari pengucapan hingga dijalankan oleh robot, serta *error* dari perintah yang diberikan terhadap robot. Analisa dan pengujian ini dilakukan dengan memberikan perintah terhadap robot sebanyak 20 kali secara acak untuk kelima perintah yang telah di tentukan :

Tabel 7. Hasil pengujian dengan respon robot

Perintah	Gerak Robot	Delay
Maju	Maju	1 detik
Kanan	Kanan	1 detik
Kiri	Kiri	1 detik
Stop	Stop	1 detik
Mundur	Mundur	1 detik
Kiri	Kiri	1 detik
Stop	Stop	1 detik
Kanan	Kanan	1 detik
Kiri	Kiri	1 detik
Mundur	Mundur	1 detik
Stop	Stop	1 detik
Kiri	Kiri	1 detik
Kiri	Kiri	1 detik
Kanan	Maju	1 detik
Stop	Stop	1 detik
Maju	Maju	1 detik
Kanan	Kanan	1 detik
Stop	Stop	1 detik
Kiri	Kiri	1 detik
Maju	Mundur	1 detik

V. Penutup

5.1 Kesimpulan

Dari hasil penelitian dan pembahasan dapat disimpulkan bahwa :

1. Semakin kompleks suatu jaringan maka semakin bagus kinerja jaringan.
2. Pada pengujian diperoleh laju pembelajaran terbaik yaitu sebesar 0,1.
3. Prosentase pengenalan yang diperoleh dengan data uji dari responden di luar *database* adalah net1 = 45,33 %, net2 = 44 %, net3 = 58 %, net4 = 44 %, net5 = 58 %, net6 = 49,33 %, net7 = 65,33 %, net8 = 65,33 %, net9 = 66,67 %. Sedangkan pengenalan dengan data uji dari responden di dalam *database* adalah net1 = 76 %, net2 = 75 %, net3 = 81 %, net4 = 75 %, net5 = 80 %, net6 = 71 %, net7 = 80 %, net8 = 82 %, net9 = 86 %.
4. Dari hasil pengujian dengan *variasi* jumlah titik pada proses FFT diperoleh bahwa prosentase pengenalan tertinggi didapatkan dengan jumlah titik sebesar 512 *point*, yaitu sebesar 74,4 %.
5. Dari hasil pengujian dengan gerak robot diperoleh waktu rata-rata yang diperlukan mulai dari proses pengenalan suara, pengiriman data hingga diterima oleh robot yaitu sekitar 1 detik.

Saran

Adapun saran yang dapat diberikan sehubungan dengan pelaksanaan penelitian ini adalah :

1. Diperlukan suatu ruangan yang cukup tenang dan bersih terhadap sinyal derau.
2. Pengenalan ucapan bisa ditambahkan dengan *filter* adaptif sehingga akan diperoleh hasil pengenalan yang tidak terpengaruh oleh derau.
3. Untuk memperbaiki kinerja jaringan, data yang dilatih perlu diperbanyak dengan menambah jumlah responden pada data pelatihan.

4. Perekaman dilakukan dengan suara responden yang asli (menggunakan pelafalan yang baik dan benar), tidak dalam keadaan dibuat-buat.
5. Pengiriman data ke robot dapat dikembangkan dengan menggunakan *bluetooth* melalui ponsel, sehingga diperoleh sistem pengendalian robot yang lebih efisien.

Daftar Pustaka

- [1] Adipranata, Rudy dan Resmana, *Pengenalan Suara Manusa dengan Metode LPC dan Jaringan Syaraf Tiruan Propagasi Balik*, Universitas Kristen Petra, Surabaya, 1999.
- [2] Bachtiar, Syafur, Irfan, *Aplikasi Pengenalan Wicara HMM untuk Kendali Robot PDA, Proyek Akhir D-3, Institut Teknologi Sepuluh November*, Surabaya, 2007.
- [3] Bayhaki, Achmad, *Penekanan Derau Secara Adaptif Pada Pengenalan Ucapan Kata*, *Skripsi S-1, Universitas Diponegoro, Semarang*, 2006.
- [4] Drs. Jong Jek Siang, M. Sc., *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab*, ANDI, Yogyakarta, 2004.
- [5] Furui, S., *Digital Speech Processing, Synthesis, and Recognition*, Marcel Dekker, Inc., New York, 1989.
- [6] Gold, B., and Morgan, N., *Speech and Audio Signal Processing : Processing and Perception of Speech and Music*, John Wiley & Sons, Inc., New York, 1999.
- [7] Hermawan, Arief, *Jaringan Syaraf Tiruan Teori dan Aplikasi*, ANDI, Yogyakarta, 2006.
- [8] Sitanggang, Doni, *Pengenalan Vokal Bahasa Indonesia dengan Jaringan Syaraf Tiruan Melalui Transformasi Fourier*, *Skripsi S-1, Universitas Diponegoro, Semarang*, 2002.

Biodata Penulis



Mahmud Irfandy, terlahir di kota Surabaya pada 05 Februari 1986. Telah menjalani pendidikan di Taman Kanak-kanak Sarinah Sidoarjo, Sekolah Dasar Negeri Wadungasri I Sidoarjo, Sekolah Lanjutan Tingkat Pertama Negeri 2 Semarang, Sekolah Menengah Umum Negeri 2 Semarang. Dan sekarang tengah menyelesaikan pendidikan Strata Satu di konsentrasi Elektronika dan Telekomunikasi, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro, Semarang, Indonesia.

Menyetujui,

Dosen Pembimbing I,

Dosen Pembimbing II,

Achmad Hidayatno, ST, MT
NIP. 132 137 933

Darjat, ST, MT
NIP. 132 231 135